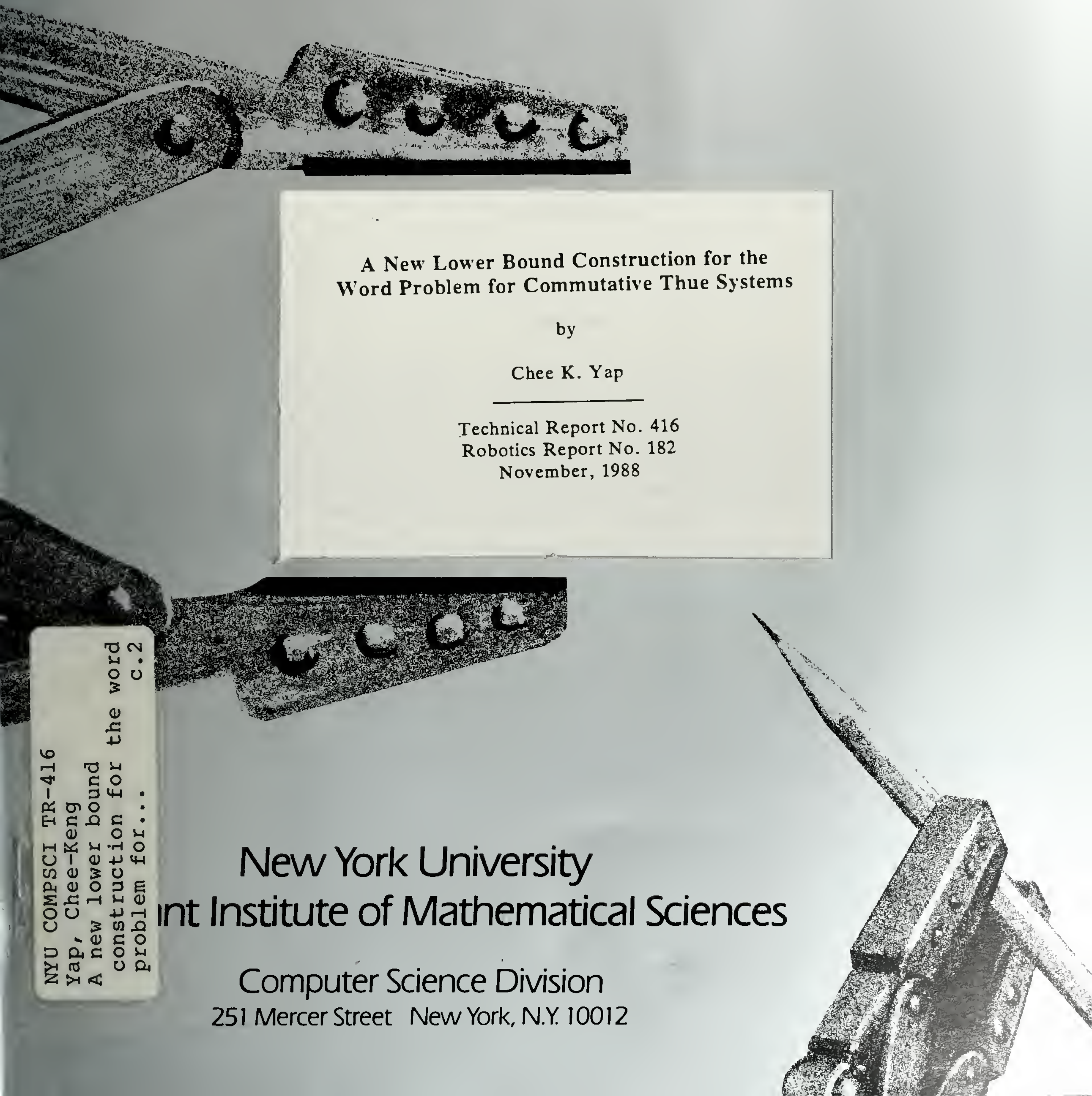
Robotics Research Technical Report



A New Lower Bound Construction for the Word Problem for Commutative Thue Systems

by

Chee K. Yap

Technical Report No. 416
Robotics Report No. 182
November, 1988

NYU COMPSCI TR-416
Yap, Chee-Keng
A new lower bound
construction for the word
problem for...
c.2

New York University
nt Institute of Mathematical Sciences

Computer Science Division
251 Mercer Street New York, N.Y. 10012

A New Lower Bound Construction for the Word Problem for Commutative Thue Systems

by

Chee K. Yap

Technical Report No. 416
Robotics Report No. 182
November, 1988

New York University
Dept. of Computer Science
Courant Institute of Mathematical Sciences
251 Mercer Street
New York, New York 10012

Work on this paper has been supported in part by NSF Grants DCR-84-01898, CCR-87-03458 and ONR Grant N00014-85-K-0046. This report is simultaneously released as a technical report of the *Research Institute of Symbolic Computation* (RISC-LINZ), Johannes Kepler University, Linz, Austria.

A New Lower Bound Construction for the Word Problem for Commutative Thue Systems

Chee K. Yap
Courant Institute of Mathematical Sciences
New York University
251 Mercer Street
New York, NY 10012

and

Fachbereich Mathematik,
Freie Universität Berlin,
Arnimallee 206,
D-1000 Berlin

Abstract

We describe a commutative Thue system that has $\sim 2n$ variables and $O(n)$ rules, each rule of length $d + O(1)$ and that counts to d^{2^n} in a certain technical sense. This gives a more efficient alternative to a well-known construction of Mayr and Meyer. Such constructions are used to derive exponential space lower bounds for the word problem for commutative Thue systems, but more recently have been used to yield double-exponential degree bounds for Gröbner bases and syzgies.

1 Introduction

Mayr and Meyer [4] proved that the uniform word problem for commutative Thue systems is exponential space complete. The key to their proof is the construction of a commutative Thue system S_M that "counts" to 2^{2^n} in this sense: there are distinguished variables A, Q_0 and Q_∞ such that if

$$Q_0 \overset{*}{\rightarrow} w \,(\mathrm{mod}\; S_M)$$

and Q_∞ occurs in w then necessarily $w = Q_\infty A^{2^{2^n}}$. (These notations are defined in the next section.) Their construction S_M has been the basis of several other lower bounds proofs in polynomial ideal theory [1][3][5].

Although the original construction of Mayr-Meyer uses $14n$ variables, various authors (including Lazard, Bayer-Stillman) have noticed that $10n$ variables suffice without any essential change. A nice exposition of the Mayr-Meyer construction has been given by Bayer-Stillman [1]. In this paper we give a more efficient alternative to the Mayr-Meyer construction, using only $\sim 2n$ variables. We remark that it is of some interest to try reduce the number of variables, especially to $\sim n$ variables, since this could lead to matching upper and lower bounds for the maximum degree of reduced Gröbner basis. The sharpest upper bound to date has been shown by Dubé [2]. In particular, Dubé proved that the reduced Gröbner basis of a set of polynomials on n variables has maximum degree bounded by d^{2^n} where d is a bound on the degree of the original polynomials.

In view of the applications of such constructions, it is important to find simple-to-understand constructions. At least the basic form of our construction (section 3, compressed rules) has this property. Our construction as well as the Meyer-Mayr construction can be viewed as the attempt to convert what is essentially a reversible nondeterministic computing device (as represented by a Thue system) into a deterministic one – such constructions go back to Post [6].

This paper is organized as follows: Section 2 establishes some terminology. In section 3, we describe a Thue system S_0 that counts up to d^{2^n}, and section 4 proves that S_0 has certain uniqueness properties. Sections 5 to 7 give successive modifications to S_0 until the final S_∞ which uses $2n + O(\sqrt{n})$ variables and has $O(n)$ rules each of size $d + O(1)$. The correctness of S_∞ is made transparent by the technique of successively "embedding" a system S_i with certain desired properties into another system S_{i+1} with the result that S_{i+1} has analogous properties. This approach also has the advantage of exposing the different ideas that go into the final Thue system. We conclude in section 8, using S_2 to show that the ideal membership and minimal syszgy problems can have double exponential degrees in the worst case.

2 Preliminaries

Let $\Sigma = \{X_1, \ldots, X_n\}$ be an alphabet where each X_i is called a (commutative) *variable*. Let $\Sigma^{\oplus}$ denote the free commutative monoid generated by Σ. An element w of $\Sigma^{\oplus}$ is called a (commutative) *word* and w can be regarded as a function

$$w : \Sigma \rightarrow \mathbf{N}$$

where $\mathbf{N}$ is the set of non-negative integers. If $X \in \Sigma$, we call $w(X)$ the *X-degree* of w and denote it by $\deg_X(w)$. We prefer to write w in the form

$$w = \prod_{i=1}^{n} X_i^{\deg_{X_i}(w)} = X_1^{\deg_{X_1}(w)} X_2^{\deg_{X_2}(w)} \cdots X_n^{\deg_{X_n}(w)}.$$

A *semi-Thue system* (over Σ) is a finite set S of pairs in $\Sigma^{\oplus}$. Each pair $(\alpha, \beta) \in S$ is called a *rule* (or production). S is called a *Thue system* if $(\alpha, \beta) \in S$ implies $(\beta, \alpha) \in S$. Call (β, α) the *reverse* of (α, β). Also β and α (respectively) are called the *precondition* and *postcondition* of the rule (α, β).

Given commutative words v, w, we write

$$v \rightarrow w (\texttt{mod } S)$$

if for some $(\alpha, \beta) \in S$ and $\gamma \in \Sigma^{\oplus}$, we have $v = \gamma\alpha$ and $w = \gamma\beta$. We also call "$v \rightarrow w$" a *transition* (of S). The reflexive, transitive closure of $\rightarrow$ ($\texttt{mod}S$) is denoted $\overset{*}{\rightarrow}$ ($\texttt{mod}S$). A sequence $(w_1, w_2, \ldots, w_k)$ ($k \geq 2, w_i \in \Sigma^{\oplus}$) is called a *derivation* of S from w_1 to w_k if $w_i \rightarrow w_{i+1}(\texttt{mod}S)$ for $i = 1, \ldots, k-1$. The derivation has a *repetition* if $w_i = w_j$ for some $i < j$; the repetition is *trivial* if $i = j - 2$. A derivation is *repetition-free* (resp., *simple*) if it has no repetition (resp., non-trivial repetition). A derivation from w_1 to w_k is said to be *unique* if it is repetition-free and there are no other repetition-free derivations from w_1 to w_k.

In this paper, we are interested in derivations from a distinguished word w_0 associated with S which we call its *initial assertion*. All words that can be derived from the initial assertion are called *assertions* of S. (This terminology is from Post[6].)

Let $\Gamma \subseteq \Sigma$ and $w, v \in \Sigma^{\oplus}$. Define $\deg_\Gamma(w)$ to be $\sum_{X \in \Gamma} \deg_X(w)$. If $\Gamma = \Sigma$, then $\deg_\Sigma(w)$ is simply denoted $\deg(w)$. The unique word $w \in \Sigma^{\oplus}$ with $\deg(w) = 0$ is denoted (without fear of confusion) by "1". We say w is *linear* in Γ if $\deg_\Gamma(w) = 1$. We say w *divides* v, denoted $w \mid v$, if for all $X \in \Sigma, \deg_X(w) \leq \deg_X(v)$. If w does not divide v, we write $w \nmid v$.

Let (S, Σ) be a commutative Thue System, $\Sigma = \{X_1, \ldots, X_n\}$. Let R denote the polynomial ring $\mathbf{Q}[\Sigma] = \mathbf{Q}[X_1, \ldots, X_n]$ where $\mathbf{Q}$ is the field of rational numbers. For each rule $\alpha \rightarrow \beta \in S$, we form the polynomial $\beta - \alpha \in R$, and let F_S denote the resulting set of polynomials.

The *word problem* (for commutative Thue systems) is this: Given (S, α, β) where α, β are commutative words over Σ, we want to know if $\alpha \overset{*}{\rightarrow} \beta(\texttt{mod}S)$. The *ideal membership problem* is this: Given (F, f) where $F \subseteq R$ is a finite set of polynomials and $f \in R$, we want to know if $f \in (F)$ where $(F) \subseteq R$ is the ideal generated by F. As shown in [4], the word problem can be reduced to the ideal membership problem as follows:

$$\alpha \overset{*}{\rightarrow} \beta(\texttt{mod}S) \text{ iff } \beta - \alpha \in (F_S).$$

Let $F \subseteq R$ and suppose we fix some arbitrary order on the elements in F, and write $\overline{F} = (f_1, \ldots, f_m)$. Then a *syzygy* of $\overline{F}$ (or simply, F) is $\overline{g} = (g_1, \ldots, g_m) \in R^m$ such that $\sum_{i=1}^{m} f_i g_i = 0$. Clearly the set of syzygies form an R-module in the natural way. The *degree* of $\overline{g}$ is $\max_{i=1,\ldots,m}\{\deg(g_i)\}$, and written $\deg(\overline{g})$. A syzygy $\overline{g}$ is *minimal* if it cannot be written as the (componentwise) sum of two syzygies, $\overline{g} = \overline{g'} + \overline{g''}$ where $\deg(\overline{g'}) > 0, \deg(\overline{g''}) > 0$ and $GCD(\overline{g}) = 1$.

3 First steps: the compressed rules

We give a simple construction of a "quadratic counter" that will subsequently be improved upon.

Constants. Throughout this construction, we fix the integers $n \geq 1$ and $d \geq 2$. We also define $e_d(k) = e(k) := d^{2^k}$ for all $k \geq 0$. Observe that $e(k+1) = e(k)^2$.

Variables. There are three types of variables.

1. *State variables:* $Q_{\texttt{start}}$, $Q_{\texttt{finish}}$
2. *Accumulator variables:* $A_k, B_k,$ $(k = 0, 1, \ldots, n)$
3. *Flag variables:* $F_k[\texttt{color}]$ $(k = 1, \ldots, n;\ \texttt{color} \in \{\texttt{inc}, \texttt{dec}, \texttt{pass}\})$.

Let Σ_0 denote this set of $5n + 4$ variables. Except for the state variables, the others each belong to some *level* which is an integer from 0 to n. We are mainly interested in commutative words of the following form:

$$w = Q A_0^{m_0} B_0^{n_0} \prod_{k=1}^{n} A_k^{m_k} B_k^{n_k} F_k[\texttt{color}_k]$$

where Q is a state variable and $m_0, n_0, m_k, n_k \geq 0$. We call such words *well-formed.*

Note in particular, a well-formed word is linear in $\{Q_{\texttt{start}}\ , Q_{\texttt{finish}}\}$ and linear in $\{F_k[\text{inc}], F_k[\text{dec}], F_k[\text{pass}]\}$ for each $k = 1, \ldots, n$.

For $1 \leq k \leq n$ and $k \leq \ell \leq n+1$, we use the abbreviation:

$$F_{k,\ell}[\texttt{color}] \equiv \prod_{i=k}^{\ell-1} F_i[\texttt{color}]$$

Thus $F_{k,k}[\texttt{color}] = 1$. The following well-formed word is designated the *initial assertion*

$$w_0 \ := \ Q_{\texttt{start}} A_0^d F_{1,n+1}[\texttt{inc}].$$

The rules which we will next specify ensure that all assertions are well-formed. Furthermore, the flag variables at each level cycle through the sequence

$$\texttt{inc} \Longrightarrow \texttt{pass} \Longrightarrow \texttt{dec} \Longrightarrow \texttt{inc} \Longrightarrow \cdots$$

We now present the rules, which naturally fall under two groups:

Start Rules. $(k = 1, 2, \ldots, n)$

$(\mathrm{S1})_k$ $\quad A_0 \xrightarrow{Q_{\texttt{start}} F_{1,k}[\texttt{pass}] F_k[\texttt{inc}]} B_0 A_k^d$ (“increment rule”)

$(\mathrm{S2})_k$ $\quad A_0 B_k^d \xrightarrow{Q_{\texttt{start}} F_{1,k}[\texttt{pass}] F_k[\texttt{dec}]} B_0$ (“decrement rule”)

$(\mathrm{S3})$ $\quad Q_{\texttt{start}} B_0^d \longrightarrow Q_{\texttt{finish}}$ (“finish rule”)

Finish Rules. $(k = 1, 2, \ldots, n-1)$

$(\mathrm{F1})_k$ $\quad Q_{\texttt{finish}} F_{1,k}[\texttt{dec}] F_k[\texttt{inc}] \longrightarrow Q_{\texttt{start}} A_0^d F_{1,k}[\texttt{inc}] F_k[\texttt{pass}]$
(“inc ⇒ pass rule”)

$(\mathrm{F2})_k$ $\quad Q_{\texttt{finish}} F_{1,k}[\texttt{dec}] A_k \xrightarrow{F_k[\texttt{pass}]} Q_{\texttt{start}} A_0^d F_{1,k}[\texttt{inc}] B_k$
(“pass ⇒ pass rule”)

$(\mathrm{F3})_k$ $\quad Q_{\texttt{finish}} F_{1,k}[\texttt{dec}] F_k[\texttt{pass}] A_k \longrightarrow Q_{\texttt{start}} A_0^d F_{1,k}[\texttt{inc}] F_k[\texttt{dec}] B_k$
(“pass ⇒ dec rule”)

Let (S_0, Σ_0) denote the Thue system corresponding to these rules.

Remark: We write the rule (α, β) as $\alpha \rightarrow \beta$ above to be suggestive of the “forward” direction of applying the rules. But one must remember that we are describing a Thue system, so the reverse rule $\beta \rightarrow \alpha$ is also implied. A derivation that only uses the forward (resp. reverse) rules will be called a *forward* (resp. *reverse*) derivation; otherwise the derivation is *mixed*. Furthermore, if $\alpha\gamma \rightarrow \beta\gamma$, we may write the rule as

$$\alpha \xrightarrow{\gamma} \beta$$

as in (S1) and (S2). (We call (S1), for instance, a rule even though it is really a family of n rules parameterized by k; this abuse of language should be harmless.) Since γ is unchanged, we call it a "catalyst" for the rule. In this section, all transitions $\rightarrow$, $\xrightarrow{*}$ are understood to be $(\mathtt{mod} S_0)$.

Let us briefly comment on the rules: the start rules are applicable only if we are in the state $Q_{\mathtt{start}}$. Rule (S1) says that the accumulator variable A_0 can convert to B_0 in the presence of (the catalyst) $Q_{\mathtt{start}} F_{1,k}[\mathtt{pass}] F_k[\mathtt{inc}]$, at the same time producing d copies of A_k. We see that the "pass" flags $F_i[\mathtt{pass}]$ at levels $i = 1$ to $i = k-1$ are used to signal that the accumulators A_i should be ignored, and the flag $F_k[\mathtt{inc}]$ signals that A_k should be "incremented". Rule (S2) is the counterpart of (S1) where the flag $F_k[\mathtt{dec}]$ signals the "decrementing" of accumulator B_k. For instance, if we start with d copies of A_0, and the catalyst for rule (S1) is present (this is the case with the initial assertion w_0), then we may apply rule (S1) d times. This converts A_0^d to B_0^d. Which brings us to rule (S3) – this rule is now applicable – and we end up in the finish state $Q_{\mathtt{finish}}$. Of course by the time we finished, we managed to create $e(1) = d^2$ copies of A_k. Conversely, if we managed to apply (S2) d times before applying (S3), we would have destroyed $e(1)$ copies of B_k.

Let us similarly note the salient features of the finish rules: in each case, the state $Q_{\mathtt{finish}}$ transforms to $Q_{\mathtt{start}} A_0^d$. Furthermore, the initial block of "decrement flags" $F_{1,k}[\mathtt{dec}]$ is converted to $F_{1,k}[\mathtt{inc}]$. Hence the differences among the finish rules lie in the flag at level k: in case of (F1), the flag is $F_k[\mathtt{inc}]$ and we convert it to $F_k[\mathtt{pass}]$. In (F2), the flag $F_k[\mathtt{pass}]$ acts only as a catalyst for converting an A_k to B_k. In (F3), we convert $F_k[\mathtt{pass}]$ to $F_k[\mathtt{dec}]$, and at the same time convert an A_k to a B_k.

Evidently, both (F2) and (F3) have the same preconditions. This introduces a non-deterministic choice in derivations. However, we shall see that this choice is forced if we want the longest possible derivation. Again, if we look at the reverse of the rules, we notice that the precondition for the reverse of rule (F2) subsumes the precondition for the reverse of (F1). We shall analyze the consequences of these non-deterministic choices later.

We see here that the flags at each level are transformed by these finish rules in the cycle ($\mathtt{inc} \Rightarrow \mathtt{pass} \Rightarrow \mathtt{dec} \Rightarrow \mathtt{inc} \Rightarrow \ldots$) as noted earlier. Of course, one must remember that these rules are reversible (we are describing a Thue System here) so this cyclic nature needs to be stated more precisely: *in any forward derivation, the flags at each level transform cyclically as indicated.*

Finally, observe that the rules preserve well-formedness of words. In particular, all assertions are well-formed.

To see these rules in action, it is best to follow the proof of the following lemma:

Lemma 1 *Let $1 \leq k \leq \ell \leq n$ and $p \geq 0$, and let w be any commutative word.*

$(\mathrm{a})_{\mathrm{k},\ell}$ *Let* $u_1 = wQ_{\mathtt{start}}A_0^dA_\ell^pF_{1,k}[\mathtt{inc}]F_{k,\ell}[\mathtt{pass}]F_\ell[\mathtt{inc}]$. *Then* $u_1 \overset{\bullet}{\to} u_1'$ *where*

$$u_1' = wQ_{\mathtt{finish}}A_\ell^{p+e(k)}F_{1,k}[\mathtt{dec}]F_{k,\ell}[\mathtt{pass}]F_\ell[\mathtt{inc}]$$

.

$(\mathrm{b})_{\mathrm{k},\ell}$ *Let* $v_1 = wQ_{\mathtt{start}}A_0^dB_\ell^{p+e(k)}F_{1,k}[\mathtt{inc}]F_{k,\ell}[\mathtt{pass}]F_\ell[\mathtt{dec}]$. *Then* $v_1 \overset{\bullet}{\to} v_1'$ *where*

$$v_1' = wQ_{\mathtt{finish}}B_\ell^pF_{1,k}[\mathtt{dec}]F_{k,\ell}[\mathtt{pass}]F_\ell[\mathtt{dec}]$$

.

Proof: First let $k = 1$.

$(\mathrm{a})_{1,\ell}$ $\quad wQ_{\mathtt{start}}A_0^dA_\ell^pF_{1,\ell}[\mathtt{pass}]F_\ell[\mathtt{inc}]$

$$\xrightarrow{(S1)} wQ_{\mathtt{start}}A_0^{d-1}B_0A_\ell^{p+d}F_{1,\ell}[\mathtt{pass}]F_\ell[\mathtt{inc}]$$

$\vdots$ (apply (S1) for $d-1$ more times)

$$\xrightarrow{(S1)} wQ_{\mathtt{start}}B_0^dA_\ell^{p+e(1)}F_{1,\ell}[\mathtt{pass}]F_\ell[\mathtt{inc}]$$

$$\xrightarrow{(S3)} wQ_{\mathtt{finish}}A_\ell^{p+e(1)}F_{1,\ell}[\mathtt{pass}]F_\ell[\mathtt{inc}]$$

$(\mathrm{b})_{1,\ell}$ $\quad wQ_{\mathtt{start}}A_0^dB_\ell^{p+e(1)}F_{1,\ell}[\mathtt{pass}]F_\ell[\mathtt{dec}]$

$$\xrightarrow{(S2)} wQ_{\mathtt{start}}A_0^{d-1}B_0B_\ell^{p+e(1)-d}F_{1,\ell}[\mathtt{pass}]F_\ell[\mathtt{dec}]$$

$\vdots$ (apply (S2) for $d-1$ more times)

$$\xrightarrow{(S2)} wQ_{\mathtt{start}}B_0^dB_\ell^pF_{1,\ell}[\mathtt{pass}]F_\ell[\mathtt{dec}]$$

$$\xrightarrow{(S3)} wQ_{\mathtt{finish}}B_\ell^pF_{1,\ell}[\mathtt{pass}]F_\ell[\mathtt{dec}]$$

Now assume $k > 1$.

$(\mathrm{a})_{k,\ell}$

$$\begin{array}{rlcl}
 & wQ_{\mathtt{start}}A_0^dA_\ell^pF_{1,k}[\mathtt{inc}]F_{k,\ell}[\mathtt{pass}]F_\ell[\mathtt{inc}] & = & u_1 \\
\xrightarrow[*]{\mathtt{ind}(a)_{k-1,k-1}} & wQ_{\mathtt{finish}}A_{k-1}^{e(k-1)}A_\ell^pF_{1,k-1}[\mathtt{dec}]F_{k-1}[\mathtt{inc}]F_{k,\ell}[\mathtt{pass}]F_\ell[\mathtt{inc}] & = & u_2 \\
\xrightarrow{(\mathrm{F1})} & wQ_{\mathtt{start}}A_0^dA_{k-1}^{e(k-1)}A_\ell^pF_{1,k-1}[\mathtt{inc}]F_{k-1}[\mathtt{pass}]F_{k,\ell}[\mathtt{pass}]F_\ell[\mathtt{inc}] & = & u_3 \\
\xrightarrow[*]{\mathtt{ind}(a)_{k-1,\ell}} & wQ_{\mathtt{finish}}A_{k-1}^{e(k-1)}A_\ell^{p+e(k-1)}F_{1,k-1}[\mathtt{dec}]F_{k-1,\ell}[\mathtt{pass}]F_\ell[\mathtt{inc}] & = & u_4 \\
\xrightarrow{(\mathrm{F2})} & wQ_{\mathtt{start}}A_0^dA_{k-1}^{e(k-1)-1}B_{k-1}^1A_\ell^{p+e(k-1)}F_{1,k-1}[\mathtt{inc}]F_{k-1,\ell}[\mathtt{pass}]F_\ell[\mathtt{inc}] & = & u_5 \\
 & & & \\
\xrightarrow[*]{\mathtt{ind}(a)_{k-1,\ell}} & wQ_{\mathtt{finish}}A_{k-1}^{e(k-1)-1}B_{k-1}^1A_\ell^{p+2e(k-1)}F_{1,k-1}[\mathtt{dec}]F_{k-1,\ell}[\mathtt{pass}]F_\ell[\mathtt{inc}] & = & u_6 \\
\vdots & (\text{by } e(k-1)-2 \text{ more applications of last 2 steps}) & & \\
\xrightarrow[*]{\mathtt{ind}(a)_{k-1,\ell}} & wQ_{\mathtt{finish}}A_{k-1}B_{k-1}^{e(k-1)-1}A_\ell^{p+e(k)}F_{1,k-1}[\mathtt{dec}]F_{k-1,\ell}[\mathtt{pass}]F_\ell[\mathtt{inc}] & = & u_7 \\
\xrightarrow{(\mathrm{F3})} & wQ_{\mathtt{start}}A_0^dB_{k-1}^{e(k-1)}A_\ell^{p+e(k)}F_{1,k-1}[\mathtt{inc}]F_{k-1}[\mathtt{dec}]F_{k,\ell}[\mathtt{pass}]F_\ell[\mathtt{inc}] & = & u_8 \\
\xrightarrow[*]{\mathtt{ind}(b)_{k-1,k-1}} & wQ_{\mathtt{finish}}A_\ell^{p+e(k)}F_{1,k-1}[\mathtt{dec}]F_{k-1}[\mathtt{dec}]F_{k,\ell}[\mathtt{pass}]F_\ell[\mathtt{inc}] & = & u_1'
\end{array}$$

Note: The inductive invocations of the lemma are labeled "ind(a)" or "ind(b)".

$(\mathrm{b})_{k,\ell}$ We omit the similar derivation for this case: the only difference is that all but the first inductive invocation turns from "ind(a)" to "ind(b)". Q.E.D.

We shall call the derivations explicitly given in the proof of this lemma the "standard derivations" (at level k). In the next section we prove that these are essentially unique.

4 Uniqueness properties

If we apply lemma 1 to the initial assertion w_0 with the parameters $k = \ell = n$, $p = 0$, then we get $w_0 \xrightarrow{*} w_\infty$ where

$$w_\infty := Q_{\mathtt{finish}}A_n^{e(n)}F_{1,n}[\mathtt{dec}]F_n[\mathtt{inc}].$$

No rule (except the reverse of rule (S3)) is now applicable to w_∞. Thus w_∞ is a *terminal* assertion, that is, if it occurs in any simple derivation starting from w_0 then it must be the last assertion in the derivation. Such a derivation can be shown to be essentially the longest simple derivation from w_0.

Let us observe the effects of using the reverse rules of the system S_0 in derivations that start from w_0. Clearly, something interesting can only happen if we apply the reverse of rule (F1) to a word that has just been derived using rule (F2):

$$\begin{aligned} wQ_{\mathtt{finish}}F_{1,k}[\mathtt{dec}]F_k[\mathtt{pass}]A_k &\xrightarrow{\text{(F2)}} wQ_{\mathtt{start}}A_0^d F_{1,k}[\mathtt{inc}]F_k[\mathtt{pass}]B_k \\ &\xrightarrow{\text{reverse(F1)}} wQ_{\mathtt{finish}}F_{1,k}[\mathtt{dec}]F_k[\mathtt{inc}]B_k \\ &\xrightarrow{\text{reverse(S3)}} wQ_{\mathtt{start}}B_0^d F_{1,k}[\mathtt{dec}]F_k[\mathtt{inc}]B_k \end{aligned}$$

If $k > 1$ then we can continue for d more steps, applying reverse of rule (S2) to finally yield

$$wQ_{\mathtt{start}}A_0^d A_1^{e(1)} F_{1,k}[\mathtt{dec}]F_k[\mathtt{inc}]B_k.$$

If $k = 1$, we can continue for at most d steps, depending on the A_1-degree. In any case, the transitions after applying reverse of (F1) are all forced transitions and they are reverse transitions. This gives us the following observation: *In any simple derivation starting from w_0, the reverse transitions (if any) occur contiguously at the end of the derivation.*

Recall that all assertions are well-formed. We now define a commutative word w to be *very well-formed* if

(1) w is well-formed.

(2) $\deg_{A_0}(w) + \deg_{B_0}(w)$ is either 0 or d.

(3) For all $k = 1, \ldots, n$,
if $F_k[\mathtt{inc}] \mid w$ then $\deg_{A_k}(w) \leq e(k), \deg_{B_k}(w) = 0$;
if $F_k[\mathtt{pass}] \mid w$ then $\deg_{A_k}(w) + \deg_{B_k}(w) = e(k)$;
if $F_k[\mathtt{dec}] \mid w$ then $\deg_{A_k}(w) = 0$, $\deg_{B_k}(w) \leq e(k)$.

We observe that the standard derivations starting from w_0 involve only very well-formed assertions. The next lemma shows that such derivations are unique under suitable conditions.

For $k = 0, 1, \ldots, n$, and any word w, let

$$\deg_k(w) = \deg_{A_k}(w) + \deg_{B_k}(w).$$

Define the *depth* of w to be the smallest k such $\deg_k(w) > 0$).

Lemma 2 (Basic Lemma) *Let $1 \leq k \leq \ell \leq n$, and $p \geq 0$ and let w be a commutative word of depth $\geq k$.*

(a) Let $u_1 = wQ_{\texttt{start}}A_0^d A_\ell^p F_{1,k}[\mathsf{inc}]F_{k,\ell}[\mathsf{pass}]F_\ell[\mathsf{inc}]$ be well-formed. Let D_1 be a simple derivation from u_1 to u_1' where u_1' is the only word in D_1 divisible by $Q_{\texttt{finish}}F_{1,k}[\mathsf{dec}]$. Then

$$u_1' = wQ_{\texttt{finish}}A_\ell^{p+e(k)}F_{1,k}[\mathsf{dec}]F_{k,\ell}[\mathsf{pass}]F_\ell[\mathsf{inc}].$$

Moreover, D_1 is unique, and the last transition in D_1 is an application of rule (S3).

(b) Let $v_1 = wQ_{\texttt{start}}A_0^d B_\ell^p F_{1,k}[\mathsf{inc}]F_{k,\ell}[\mathsf{pass}]F_\ell[\mathsf{dec}]$ be well-formed with $\mathsf{deg}_{B_\ell}(w) = 0$. Let D_2 be a simple derivation from v_1 to v_1' where v_1' is the only word in D_2 divisible by $Q_{\texttt{finish}}F_{1,k}[\mathsf{dec}]$. Then $p \geq e(k)$ and

$$v_1' = wQ_{\texttt{finish}}B_\ell^{p-e(k)}F_{1,k}[\mathsf{dec}]F_{k,\ell}[\mathsf{pass}]F_\ell[\mathsf{dec}].$$

Moreover, D_2 is unique, and the last transition in D_2 is an application of rule (S3).

Proof

First let $k = 1$. We shall be referring to the "standard derivation" as used in the proof of lemma 1.

(a) Since the depth of w is $\geq k \geq 1$, $\mathsf{deg}_{B_0}(u_1) = 0$. Moreover, since u_1 is well-formed, w contains no flag variables at level $\leq \ell$. Thus the first applicable rule $(\mathrm{S1})_\ell$ in the standard derivation is forced. The next $d-1$ applications of (S1) is also forçed (since we allow only simple derivations. Then the application of (S3) is also forced. Call the result u_1''. Now $Q_{\texttt{finish}}F_{1,1}[\mathsf{dec}] = Q_{\texttt{finish}}$ divides u_1''. It is now evident that $u_1'' = u_1'$ and has the form stipulated in the lemma.

(b) The proof is similar.

Now assume $k > 1$

(a) Since $Q_{\texttt{finish}}F_{1,k}[\mathsf{dec}] \mid u_1'$, there must be a first word u_1'' in the derivation D_1 in which $Q_{\texttt{finish}}F_{1,k-1}[\mathsf{dec}] \mid u_1''$. By the induction hypothesis (a), u_1'' has the form of u_2 in the proof of lemma 1:

$$u_2 = wQ_{\texttt{finish}}A_{k-1}^{e(k-1)}F_{1,k-1}[\mathsf{dec}]F_{k-1}[\mathsf{inc}]F_{k,\ell}[\mathsf{pass}]F_\ell[\mathsf{inc}].$$

Now the only rule applicable to u_2 is (F1) which yields u_3 as in the standard derivation. (Note that the reverse of rule (S3) is also applicable to u_2, but the induction hypothesis (a) implies that this would make D_1 non-simple.) Since $Q_{\mathtt{finish}}F_{1,k-1}[\mathtt{dec}] \mid u_1'$ but not u_3, we again see that there is a first word $u_1^{(4)}$ after u_3 in D_1 such that $Q_{\mathtt{finish}}F_{1,k-1}[\mathtt{dec}] \mid u_1^{(4)}$. By the induction hypothesis (a), $u_1^{(4)}$ must have the form u_4 as in the standard derivation. Now, we see that either rule (F2) or (F3) is applicable to u_4 (the reverse of rule (S3) is applicable but excluded by repetition-freeness).

If (F2) were applied to u_4 we get u_5 as in the standard derivation. Of course, we can now apply the reverse of (F1) although we already saw that this will not lead to anything interesting, and certainly not to u_1'. Hence (by induction hypothesis (a)) we might as well assume that (F2) is followed by the standard derivation at level $k-1$. In general, suppose we apply (F2) followed by the standard derivation at level $k-1$ for a total of $m \geq 0$ times to u_4. The final result is:

$$u_1^{(5)} = wQ_{\mathtt{finish}}A_{k-1}^{e(k-1)-m}B_{k-1}^{m}A_{\ell}^{(m+1)e(k-1)}F_{1,k-1}[\mathtt{dec}]F_{k-1,\ell}[\mathtt{pass}]F_{\ell}[\mathtt{inc}].$$

Clearly $m \leq e(k-1)$. However, if $m = e(k-1)$ then we are stuck since no rule applies. So we assume $m \leq e(k-1)-1$. Only rule (F3) is applicable to $u_1^{(5)}$, yielding

$$u_1^{(6)} = wQ_{\mathtt{start}}A_0^{d}A_{k-1}^{e(k-1)-m-1}B_{k-1}^{m+1}A_{\ell}^{(m+1)e(k-1)}F_{1,k-1}[\mathtt{inc}]F_{k-1}[\mathtt{dec}]F_{k,\ell}[\mathtt{pass}]F_{\ell}[\mathtt{inc}].$$

Again there must be a first $u_1^{(7)}$ in D_1 after $u_1^{(6)}$ where $Q_{\mathtt{finish}}F_{1,k-1}[\mathtt{dec}] \mid u_1^{(7)}$. By induction hypothesis (b), $m+1 \geq e(k-1)$ and hence $m+1 = e(k-1)$. This shows that $u_1^{(5)} = u_7, u_1^{(6)} = u_8$ and $u_1^{(7)} = u_1'$. By induction hypothesis (b), the last transition in the derivation $u_1^{(6)} \overset{*}{\to} u_1'$ is an application of rule (S3), and hence the same holds for D_1.

(b) This proof is similar to part (a). Q.E.D.

5 Expanded rules

The size of rule (α, β) is $\max\{\deg(\alpha), \deg(\beta)\}$. The rules (α, β) in the Thue system S_0 can have *size* linear[1] in n, i.e.,

$$|\alpha| = d + \Omega(n) \quad \text{or} \quad |\beta| = d + \Omega(n).$$

In this section, we convert S_0 to an essentially equivalent system S_1 whose rules have constant sizes, i.e.,

$$|\alpha| = d + O(1)\ , \ \ |\beta| = d + O(1).$$

---

[1] We deem the size dependence on d as natural: in fact, d is usually viewed as the maximum degree of the rules.

To achieve this, we introduce $n+1$ new variables

$$L_0, L_1, L_2, \ldots, L_n$$

called *level variables*. Let $\Sigma_1 = \Sigma_0 \bigcup \{L_0, \ldots, L_n\}$. The start rules (S1) and (S2) are replaced by:

$$\begin{array}{llcll}
\text{(T1)} & L_0 & \xrightarrow{Q_{\mathtt{start}}A_0} & L_1 & \\
\text{(T2)} & L_k & \xrightarrow{Q_{\mathtt{start}}F_k[\mathtt{pass}]A_0} & L_{k+1} & (k = 1, \ldots, n-1) \\
\text{(T3)} & L_kA_0 & \xrightarrow{Q_{\mathtt{start}}F_k[\mathtt{inc}]} & L_0B_0A_k^d & (k = 1, \ldots, n) \\
\text{(T4)} & L_kB_k^d & \xrightarrow{Q_{\mathtt{start}}F_k[\mathtt{dec}]} & L_0B_0 & (k = 1, \ldots, n)
\end{array}$$

Clearly, the rules (S1) and (S2) can be simulated by the new rules (T1)-(T4). For instance, rule (S1) is simulated by the sequence

$$L_0A_0 \xrightarrow[(T1)]{Q_{\mathtt{start}}} L_1A_0 \xrightarrow[(T2)]{Q_{\mathtt{start}}F_1[\mathtt{pass}]} L_2A_0 \xrightarrow[(T2)]{} \cdots \xrightarrow[(T2)]{} L_kA_0 \xrightarrow[(T3)]{Q_{\mathtt{start}}F_k[\mathtt{inc}]} A_k^dB_0L_0.$$

Next we convert rule (S3) to

$$\text{(T5)} \qquad Q_{\mathtt{start}}L_0B_0^d \longrightarrow Q_{\mathtt{finish}}L_1.$$

Similarly, the finish rules (F1)-(F3) becomes, for $k = 1, 2, \ldots, n-1$:

$$\begin{array}{llcl}
\text{(G1)} & L_kF_k[\mathtt{dec}] & \xrightarrow{Q_{\mathtt{finish}}} & L_{k+1}F_k[\mathtt{inc}] \\
\text{(G2)} & Q_{\mathtt{finish}}L_kF_k[\mathtt{inc}] & \longrightarrow & Q_{\mathtt{start}}L_0A_0^dF_k[\mathtt{pass}] \\
\text{(G3)} & Q_{\mathtt{finish}}L_kA_k & \xrightarrow{F_k[\mathtt{pass}]} & Q_{\mathtt{start}}L_0A_0^dB_k \\
\text{(G4)} & Q_{\mathtt{finish}}L_kF_k[\mathtt{pass}]A_k & \longrightarrow & Q_{\mathtt{start}}L_0A_0^dF_k[\mathtt{dec}]B_k
\end{array}$$

Again, each of the finish rules can be simulated by (G1)-(G4). For instance, (F2) becomes, for $k \geq 2$:

$$\begin{array}{rcl}
Q_{\mathtt{finish}}L_1F_{1,k}[\mathtt{dec}]F_k[\mathtt{pass}]A_k & \xrightarrow{(G1)} & Q_{\mathtt{finish}}L_2F_1[\mathtt{inc}]F_{2,k}[\mathtt{dec}]F_k[\mathtt{pass}]A_k \\
& \vdots & (k-2) \text{ more applications of (G1)} \\
& \xrightarrow{(G1)} & Q_{\mathtt{finish}}L_kF_{1,k}[\mathtt{inc}]F_k[\mathtt{pass}]A_k \\
& \xrightarrow{(G2)} & Q_{\mathtt{start}}L_0A_0^dF_{1,k}[\mathtt{inc}]F_k[\mathtt{pass}]B_k
\end{array}$$

Let the Thue system S_1 over Σ_1 refer to the set of rules (T1)–(T5), (G1)–(G4) and their reverses. Evidently, we intend well-formed words of S_1 to be linear in $\{L_0, L_1, \ldots, L_n\}$. We define a word to $w \in \Sigma_1^{\oplus}$ to be *(very) well-formed* if and only if $w = L_k w'$ for some $k = 0, \ldots, n$ and $w' \in \Sigma_0^{\oplus}$ is (very) well-formed in the original sense. If $w \in \Sigma_1^{\oplus}$ is a well-formed word such that either

$$Q_{\mathtt{start}} L_0 \mid w \qquad \text{or} \qquad Q_{\mathtt{finish}} L_1 \mid w$$

then we call w a *standard* word.

Clearly, there is a one-one map ϕ that takes each well-formed word $w' \in \Sigma_0^{\oplus}$ to a standard word $\phi(w') \in \Sigma_1^{\oplus}$. For instance, recall the assertions w_0 and w_∞ from the previous section. Then $\phi(w_0)$ is the initial assertion of S_1 and we have:

$$\phi(w_0) := Q_{\mathtt{start}} L_0 A_0^d F_{1,n+1}[\mathsf{inc}]$$

$$\phi(w_\infty) := Q_{\mathtt{finish}} L_1 A_n^{e(n)} F_{1,n}[\mathsf{dec}] F_n[\mathsf{inc}]$$

Definition. Given two Thue systems (S, Σ) and (S', Σ'), we say that S is *embedded in* S' via a map ϕ if ϕ is a one-one function from the assertions of S to the assertions of S' such that the initial assertion w_0 of S is mapped to the initial assertion w_0' of S' by ϕ, and the following holds. First, call an assertion of S' *standard* if it is of the form $\phi(w)$ for some w. Then:

(a) If $u \to w$ is a transition of S where u is an assertion, then there is a corresponding *unique* derivation

$$D' = (\phi(u), w_1', \ldots, w_m', \phi(w))$$

of S' where $w_1', \ldots, w_m'$ are all non-standard words.

(b) Conversely, if $D' = (\phi(u), w_1', \ldots, w_m', \phi(w))$ is a derivation of S' where u is an assertion of S and $w_1', \ldots, w_m'$ are non-standard words then $u \to w$ is a transition of S.

We then have the following result.

Lemma 3 *The system S_0 from the previous section is embedded in the system S_1 in this section, via the map ϕ defined above.*

We omit the easy proof. In a natural way, all the properties (such as uniqueness) we have proved for S_0 transfers to S_1.

6 Winding down, cleaning up

The system S_1, with some trivial modifications, counts to $e(n)$ in the sense of the introduction. In applications, we also want the reverse process, to "count" down from $e(n)$. It is convenient to further modify the system so that the initial assertion is a single variable Q_0 and for the standard derivation (by definition) to eventually yield a single variable Q_∞.

By winding down here, we mean the addition of rules which allow the assertion

$$\phi(w_\infty) = Q_{\mathtt{finish}} L_1 A_n^{e(n)} F_n[\mathtt{inc}] F_{1,n}[\mathtt{dec}]$$

to get rid of all the A_n's and to reach some definite word w'_∞. This is easy to do in S_1. Recall that the finish rules (G1)-(G4) do not apply for $k = n$; the new rules cover this case.

Winding Down Rules

$$\begin{array}{lll} \text{(W1)} \quad F_n[\mathtt{inc}] & \xrightarrow{\quad L_n \quad} & F_n[\mathtt{pass}] \\ \text{(W2)} \quad A_n & \xrightarrow{L_n F_n[\mathtt{pass}]} & B_n \\ \text{(W3)} \quad Q_{\mathtt{finish}} L_n F_n[\mathtt{pass}] & \xrightarrow{\qquad\qquad} & Q_{\mathtt{start}} L_0 A_0^d F_n[\mathtt{dec}] \end{array}$$

These additional rules allow the standard derivation $\phi(w_0) \xrightarrow{*} \phi(w_\infty)$ to be extended as shown here:

$$\begin{array}{lll} \phi(w_\infty) & \xrightarrow{*} & Q_{\mathtt{finish}} L_n F_{1,n}[\mathtt{inc}] A_n^{e(n)} F_n[\mathtt{inc}] \quad \text{(by } n-1 \text{ applications of (G1))} \\ & \xrightarrow{\text{(W1)}} & Q_{\mathtt{finish}} L_n F_{1,n}[\mathtt{inc}] A_n^{e(n)} F_n[\mathtt{pass}] \\ & \vdots & \text{(apply (W2) for } e(n) \text{ times)} \\ & \xrightarrow{\text{(W2)}} & Q_{\mathtt{finish}} L_n F_{1,n}[\mathtt{inc}] B_n^{e(n)} F_n[\mathtt{pass}] \\ & \xrightarrow{\text{(W3)}} & Q_{\mathtt{start}} L_0 A_0^d F_{1,n}[\mathtt{inc}] B_n^{e(n)} F_n[\mathtt{dec}] \end{array}$$

From this last word, by the analogue of lemma 1(b), we eventually derive

$$w'_\infty = Q_{\mathtt{finish}} L_1 F_{1,n+1}[\mathtt{dec}].$$

Moreover, by the usual arguments, this derivation from $\phi(w_0)$ to w'_∞ is unique.

Next, to allow the system to have unique starting and final variables, we set

$$\Sigma_2 = \Sigma_1 \bigcup \{Q_0, Q'_0, Q_\infty Q'_\infty\}$$

and introduce the following rules.

Initialization Rules

$$\begin{array}{llll}
(\mathrm{I1}) & Q_0 & \longrightarrow & Q'_0 L_1 A_0^d \\
(\mathrm{I2}) & L_k Q'_0 & \longrightarrow & Q'_0 L_{k+1} F_k[\mathtt{inc}] \qquad (k = 1, \ldots, n-1) \\
(\mathrm{I3}) & L_n Q'_0 & \longrightarrow & Q_{\mathtt{start}} L_0 F_n[\mathtt{inc}]
\end{array}$$

Clearly these rules imply $Q_0 \xrightarrow{*} \phi(w_0)$. Similarly the next set of rules imply $w'_\infty \xrightarrow{*} Q_\infty$.

Cleaning Up Rules

$$\begin{array}{llll}
(\mathrm{C1}) & Q_{\mathtt{finish}} L_1 & \longrightarrow & Q'_\infty L_1 \\
(\mathrm{C2}) & Q'_\infty L_k F_k[\mathtt{dec}] & \longrightarrow & Q'_\infty L_{k+1} \qquad (k = 1, \ldots, n-1) \\
(\mathrm{C3}) & Q'_\infty L_n F_n[\mathtt{dec}] & \longrightarrow & Q_\infty
\end{array}$$

Note that rule (C1) if applied prematurely (i.e. when not applied to w'_∞) will get 'stuck' but in any case will not yield Q_∞.

Call the Thue system just described (S_2, Σ_2). The rules of S_2 are explicitly given by the rules

(T1)-(T5), (G1)-(G4), (W1)-(W3), (I1)-(I3) and (C1)-(C3).

We saw that the following repetition-free derivation of S_2 is unique:

$$Q_0 \xrightarrow{*} Q_\infty$$

The number of variables in Σ_2 is $6n + O(1)$. However, we can easily reduce this to $4n + O(1)$ by observing that the four variables

$$F_k[\mathtt{inc}], F_k[\mathtt{pass}], F_k[\mathtt{dec}], L_k$$

at level k can easily be encoded by introducing two new variables F_k, G_k and using the 6 words $F_k^i G_k^{5-i}$ for $i = 0, \ldots, 5$: for instance L_k is either present or not in a well-formed word, and this can be encoded by the parity (even or odd) of i, and the three colors can be associated with the values $\lfloor i/2 \rfloor = 0, 1, 2$. The modifications to the rules of S_2 are trivial. Call this system (S_3, Σ_3).

7 A technique for encoding flag systems

Let Σ be a set of commutative variables and S a Thue system over Σ. We say S has *parameters* (b, n, d) if S has $\leq bn$ rules and each rule has size $\leq d+b$. (Note: previously, b has been hidden in the big-Oh notation.) We call a subset $\Gamma \subseteq \Sigma$ a *system of flags* for S (relative to (b, n, d)) provided:

(1) Each rule $(\alpha, \beta) \in S$ satisfies $\deg_{\Gamma}(\alpha\beta) < b$.

(2) Each assertion w of S satisfies $\deg_F(w) < b$ for each $F \in \Gamma$.

In this section we show this general result:

Theorem 1 *There is a universal constant $c > 0$ such that the following holds. Let S over Σ have parameters (b, n, d) and $\Gamma \subseteq \Sigma$ is a system of flags for S. Then S can be embedded in a Thue system S' over Σ' where S' has parameters (cb, n, d) and*

$$\Sigma' = (\Sigma - \Gamma) \cup \Gamma'$$

and $|\Gamma'| < cb^3\sqrt{|\Gamma|}$.

In other words, we can replace the system of flags Γ with a new set Γ' that is roughly square-root the original size.

We can apply this result to the system S_2 that has parameters (b, n, d) for some small $b < 10$. The set

$$\Gamma_1 = \{F_k[\mathrm{inc}], F_k[\mathrm{pass}], F_k[\mathrm{dec}] \;:\; k = 1, \ldots, n\}$$

is a system of flags for S_2. But so is $\Gamma_2 = \{L_1, \ldots, L_n\}$, and the union of two systems of flags is also a system of flags (perhaps using a different b constant). Applying this theorem to $\Gamma = \Gamma_1 \cup \Gamma_2$, we obtain a system (call it) S_∞ over a set of $2n + O(\sqrt{n})$ variables, such that S_2 is embedded in it. If S_2 is embedded in S_∞ via the map ϕ then the derivation

$$\phi(Q_0) \xrightarrow{*} \phi(Q_\infty) \pmod{S_\infty}$$

is unique. With trivial modifications to S_∞ to satisfy the concept of counting as formalized in the introduction, we may conclude the main result of this paper:

Theorem 2 *There is a constant $c_0 > 0$ such that for all $n \geq 1, d \geq 2$, there exists a commutative Thue system $(S_{n,d}, \Sigma_{n,d})$ such that $S_{n,d}$ counts to $e_d(n) = d^{2^n}$, $|\Sigma_{n,d}| \leq 2n + c_0\sqrt{n}$, $|S_{n,d}| \leq c_0 n$, each rule in $S_{n,d}$ has size $\leq d + c_0$.*

In the rest of this section, we will prove theorem 1. It is simpler to assume that $|\Gamma| + 1$ is a perfect square,

$$|\Gamma| + 1 = r^2$$

for some integer $r > 0$. We define the integer functions μ, ν where for any integer k,

$$\mu(k) = \left\lfloor \frac{k}{r} \right\rfloor,$$

$$\nu(k) = k \bmod r.$$

We shall use these functions for $k = 1, \ldots, |\Gamma|$; in this case $k = \mu(k)r + \nu(k)$.

Encoding of flag information by groups. Introduce $2(r+4)$ variables

$$D'_0, D''_0, \ldots, D'_{r+3}, D''_{r+3}$$

to encode the information of the flag variables Γ. We partition Γ into r *groups*, $\Gamma = \Gamma_1 \cup \Gamma_2 \cup \cdots \cup \Gamma_r$ with

$$\Gamma_i = \{F_{(i-1)r+1}, F_{(i-1)r+2}, \ldots, F_{ir}\}$$

where

$$\Gamma = \{F_1, F_2, \ldots, F_{|\Gamma|}\}.$$

Let $v \in \Gamma_i^{\oplus}$ and for each $F_{(i-1)r+j} \in \Gamma_i$, let the $F_{(i-1)r+j}$-degree of v be n_j. We assume that $n_j < b$ where b is the constant in the statement of theorem 1. We encode such a v by a word in $\{D'_i, D''_i\}^{\oplus}$ as follows. Let

$$n(v) = \sum_{j=1}^{r} n_j b^{j-1}.$$

(Remark: we shall call n_1 the zero-th (lowest) digit of the integer $n(v)$, n_2 the first digit, and so on.) Notice that $n(v) < b^r$ and so we can encode v by the word

$$D_i[n(v)] = (D'_i)^{n(v)}(D''_i)^{b^r - n(v)}.$$

The embedding map ϕ. Now if $v_1 v_2 \cdots v_r w$ is an assertion of S where $\deg_\Gamma(w) = 0$ and $v_i \in \Gamma_i^{\oplus}$ then we define the map ϕ by

$$\phi(v_1 \cdots v_r w) = Q_{\mathbf{standard}} D_1[n_1] \cdots D_r[n_r] w$$

where $Q_{\mathbf{standard}}$ is a new state variable, $n_i = n(v_i)$ is the integer encoding the degrees of the variables in Γ_i, obtained as in $n(v)$ above. We call a word of the form $\phi(w)$ a *standard word* of S' if w is an assertion of S. This function ϕ will embed S in the system S' to be constructed.

Each pair $\{D_i', D_i''\}$ forms what will be referred to as the $D_i[\,]$-counter. Such a counter encodes an integer $0 \leq m \leq b^r$ as $(D_i')^m (D_i'')^{b^r - m}$. Thus it is easy to detect if the value of such a counter is zero: we use a b-ary counter that counts up to b^r. More on b-ary counters below. It is important to realize that the map ϕ is one-one only for those words w which satisfy $\deg_F(w) < b$ for each $F \in \Gamma$; by definition of flag systems, this is the case when w is an assertion. Notice that the $D_h[\,]$-counters ($h = 0, r+1, \ldots, r+3$) are unused at this point.

The transformed rules. For each rule (R) of S we introduce a transformed rule (R)′ of S' as follows. Let (R) have the form

$$\text{(R)} \quad \sigma\gamma \rightarrow \tau\delta$$

where $\sigma\tau \in \Gamma^{\oplus}$ and $\deg_\Gamma(\gamma\delta) = 0$. Then its transformation (R)′ is:

$$\text{(R)}' \quad Q_{\mathtt{standard}}\gamma \rightarrow Q_{\mathtt{step}}\delta E[\sigma, \tau]$$

where $Q_{\mathtt{step}}$ is a new state variable and $E[\sigma, \tau]$ is a special encoding of (σ, τ) to be described next. The state $Q_{\mathtt{step}}$ in (R)′ indicates that we are now simulating one transition step of S.

The encoding $E[\sigma, \tau]$. Clearly $\deg(\sigma\tau) < b$. So σ, τ has the form

$$\sigma = F_{h_1}^{m_1} F_{h_2}^{m_2} \cdots F_{h_p}^{m_p}$$

$$\tau = F_{h_1}^{n_1} F_{h_2}^{n_2} \cdots F_{h_p}^{n_p}$$

where $1 \leq h_1 < h_2 < \cdots < h_p \leq |\Gamma|$, $\sum_{i=1}^{p} m_i + n_i < b$ and each $m_i + n_i \geq 1$. Let us note that the triple (m_i, n_i, h_i) in this context amounts to the following 'instruction'

$Z^{(m_i, n_i)}[h_i]$: 'the F_{h_i}-degree of the current word must be at least m_i for the application of the rule (R), and this F_{h_i}-degree should be incremented by $n_i - m_i$ after the application of rule (R)'.

To encode such instructions, we introduce $2(b+1)^3 r$ new variables

$$M_{i,0}^{(j,j')}, M_{i,1}^{(j,j')}, \ldots, M_{i,r-1}^{(j,j')}$$

$$N_{i,0}^{(j,j')}, N_{i,1}^{(j,j')}, \ldots, N_{i,r-1}^{(j,j')}$$

for $i, j, j' = 0, \ldots, b$. For any $k = 1, \ldots, |\Gamma|$, and $i, j, j' = 0, \ldots, b$, let $M_i^{(j,j')}[k]$ and $N_i^{(j,j')}[k]$ be shorthand for the variables

$$M_i^{(j,j')}[k] \;:=\; M_{i,\mu(k)}^{(j,j')},$$

$$N_i^{(j,j')}[k] \ := \ N_{i,\nu(k)}^{(j,j')}.$$

Also, let

$$MN_i^{(j,j')}[k] \ := \ M_i^{(j,j')}[k]N_i^{(j,j')}[k].$$

We intend for the word $MN_i^{(j,j')}[k]$ to encode the instruction $Z^{(j,j')}[k]$ (the role of i will be clearer below).

The encoding of σ, τ can now be given.

$$E[\sigma,\tau] \ := \ MN_{p+1}^{(b,b)}[1]\left(\prod_{i=1}^{p} MN_i^{(m_i,n_i)}[h_i]\right).$$

The presence of $MN_{p+1}^{(b,b)}[1]$ simply encodes the value of p in the definition of σ, τ; this encoding makes sense since $p < b$ and $m_i + n_i < b$.

Remark. Note that rule (R)′ may be applicable to a word $\phi(w)$ even though rule (R) is not applicable to w (this happens when $\gamma \neq 1$ in rule (R)). It will turn out that our rules ensure that such unintended transitions will eventually become "stuck", i.e., be unable to return to a standard word without repetition. On the other hand, if rule (R) were applicable to w, the derivation will eventually return to the standard word $\phi(w')$ where $w \ \rightarrow w'$ is a transition of S obtained by applying rule (R).

Simulation by stages. We carry out the simulation of a step in *stages* (in fact there are p stages if the rule to be simulated has σ, τ as above). At the beginning of each stage, the current word will in general contain the catalyst word

$$MN_{i+1}^{(b,b)}[1]MN_i^{(j,j')}[k] \tag{1}$$

for some i, j, j', k, indicating that

(a) We are in stage i

(b) We want to carry out instruction $Z^{(j,j')}[k]$.

When done, we proceed to the next stage by transforming $MN_{i+1}^{(b,b)}[1]MN_i^{(j,j')}[k]$ to $MN_i^{(b,b)}[1]$. When we finally reach reach stage 0 as indicated by the presence of $MN_1^{(b,b)}[1]$, we know that we had successfully checked and transformed all the flag values necessary for the application of rule (R). We may change the state from $Q_{\mathbf{step}}$ back to $Q_{\mathbf{standard}}$.

In the rest of this simulation, we assume that the integers i, j, j', k have been fixed by the occurence of the catalyst (1). Let us recall that the F_k-degree of w is encoded as the $\nu(k)^{\text{th}}$ digit of the $D_{\mu(k)}$-degree in $\phi(w)$. If m is this $D_{\mu(k)}$-degree, then the F_k-degree is given by

$$\left\lfloor \frac{m}{b^{\nu(k)}} \right\rfloor (\bmod\ b).$$

Each stage consists of seven phases.

Phase 0. *Initialization.*
We initialize the auxilliary counters $D_h[\ ]$-counters for $h = r+1, r+2, r+3$ to zero. We transfer the contents of the $D_{\mu(k)}[\ ]$-counter to the $D_0[\ ]$-counter and set up appropriate "level variables" to indicate that the b-ary counter is to be $\nu(k)$ digits long only. This ensures that the rules of the remaining phases (except for phase 6) are independent[2] of $\mu(k)$ and $\nu(k)$.

Note on b-ary counters. The reader should realize that the ability to initialize counters and transfer the contents of a counter to another depends the fact that we can set up b-ary counters that 'count' to b^r (recall the introductory section of this paper). Since these b-ary counters count to at most b^r, they are are easily implemented using $O(br)$ variables. Associated with the b-ary counter are $O(br)$ rules to increment, decrement or test-for-zero. Each such rule has size $O(1)$. When done carefully (e.g., by introducing state variables to indicate that the b-ary counter should be incremented, etc), these rules can also be shared by all the phases. Hence, we only need one b-ary counter. The details of such a construction are left as an exercise.

Phase 1. *Computes some $\ell \leq \lfloor m/b^{\nu(k)} \rfloor$ in the $D_{r+1}[\ ]$-counter, and decrements the $D_0[\ ]$-counter by $\ell b^{\nu(k)}$.*
Recall that m (the $D_{\mu(k)}$-degree of w) is encoded by the word

$$D_0[m] = (D_0')^m (D_0'')^{b^r - m}.$$

In this phase, we will reduce the $D_0[\ ]$-counter by some $\ell \geq 0$ multiples of $b^{\nu(k)}$, and store the value of ℓ in the previously unused counter $D_{r+1}[\ ]$. This is easily done, using a b-ary counter that counts to $b^{\nu(k)}$, where each increment of the b-ary counter is coupled with a decrement of the $D_0[\ ]$-counter. Whenever the b-ary counter returns to 0, we have the choice of going to the next phase or of decrementing $D_0[\ ]$-counter by another $b^{\nu(k)}$; in the latter case, we increment the value of ℓ in the $D_{r+1}[\ ]$-counter. If ℓ exceeds $\lfloor m/b^{\nu(k)} \rfloor$, we become stuck in a state from which there is no return to a standard word.

Phase 2. *Transfers the residual value of the $D_0[\ ]$-counter to the $D_{r+2}[\ ]$-counter.*
We want to check that ℓ is exactly equal to $\lfloor m/b^{\nu(k)} \rfloor$. This amounts to checking that the $D_0[\ ]$-counter has a residual value less than $b^{\nu(k)}$. So we use the b-ary counter of phase 1 to continue decrementing the $D_0[\ ]$-counter. To save the residual value of the $D_0[\ ]$-counter, we simultaneously increment the $D_{r+2}[\ ]$-counter, which is unused until now. Before each decrement of the $D_0[\ ]$-counter, we have the choice of continuing to decrement or to go to phase 3. If we succeeded in decrementing $b^{\nu(k)}$ times, then we enter a stuck state (ℓ is too small).

---

[2] If the phases are not made independent of $\mu(k)$ and $\nu(k)$ then the number of rules simply grows by a factor of r.

Phase 3. *Detects if the transfer to $D_{r+2}[\ ]$-counter is complete and thus confirms that $\ell = \lfloor m/b^{\nu(k)} \rfloor$.*
To do this, we simply verify that the $D_0[\ ]$-counter is zero (otherwise we should get stuck).

Phase 4. *Computes some $\ell' \le \lfloor \ell/b \rfloor$ in $D_{r+3}[\ ]$-counter and decrements ℓ in the $D_{r+1}[\ ]$-counter by $\ell' b$.*
We now have the value of $\ell = \lfloor m/b^{\nu(k)} \rfloor$ in the $D_{r+1}[\ ]$-counter. The F_k-degree we seek is $\ell \bmod b$. Similar to phase 1, we now subtract some $\ell' \ge 0$ multiples of b from the $D_{r+1}[\ ]$-counter, and keep track of ℓ' in the (hitherto unused) $D_{r+3}[\ ]$-counter. If ℓ' gets larger than $\lfloor \ell/b \rfloor$, we get stuck. Otherwise, we reach phase 5.

Phase 5. *Checks that $\ell' = \lfloor \ell/b \rfloor$ and computes the F_k-degree*
Again, we must check if $\ell' = \lfloor \ell/b \rfloor$. This is accomplished in one step by subtracting some integer $j'' = 0, \ldots, b-1$ (nondeterministically) from the $D_{r+1}[\ ]$-counter and then checking if the $D_{r+1}[\ ]$-counter is now zero. If the counter is indeed zero, then j'' is the F_k-degree. Otherwise we get stuck.

Phase 6. *Checks that F_k-degree is sufficient for applying the current rule and carries out the transformation of the F_k-degree.*
Now we check that j'' discovered in the previous phase is indeed $\ge j$, and if so, we make the value of the $D_{r+1}[\ ]$-counter equal to $j'' + (j' - j)$. We must now restore the value of the $D_{\mu(k)}[\ ]$-counter, using this formula

$$d_{\mu(k)} := (d_{r+3} b + d_{r+1}) \cdot b^{\nu(k)} + d_{r+2}$$

where d_h refers to the value in the $D_h[\ ]$-counter. The details are left to the reader.

This completes the description of the simulation of a transition step. Let us tabulate the statistics of the system S'.

Variables. There are $O(b^3 r)$ new variables explicitly described above. We need another $O(br)$ variables to implement the b-ary counters. Of course, there are also $O(1)$ state variables to remember each part of each phase.

Rules. We have one transformed rule for each of the rule of S; these rules have size at most $d+3b+O(1)$ (cf. (R)$'$ above). We introduced rules for the 7 phases. In phases 0 and 6, we need to transfer values between the $D_0[\ ]$-counter and the appropriate $D_i[\ ]$-counter. This can be done with $O(r)$ rules of size $O(1)$. Not counting the rules for the b-ary counters, most of the remaining rules amount to incrementing or decrementing the $D_h[\ ]$-counters for $h = 0, r+1, \ldots, r+3$; there are $O(1)$ such rules, each of size $O(1)$. In phases 5, we have b rules of size $O(b)$, corresponding to guessing the value of j''. Finally, the number of rules for the b-ary counter and their sizes are $O(br)$ and $O(1)$, respectively, as already explained.

This completes the proof of theorem 1.

Final remarks. We have not attempted to optimize the constant factors in this construction. For example, we note that the $O(b^3 r)$ MN-variables in turn form a flag system for the system S', so the theorem 1 can be applied again. Hence we could actually get away with only $|\Gamma'| = O(br)$ new variables ($2r$ variables for the $D_i[\ ]$-counters and $O(br)$ for the binary counters). Another idea that can improve constant factors is that some flag systems (such as the level variables $L_1, \ldots, L_n$) actually satisfy the stronger property that each assertion has bounded degree in Γ, not just bounded degree in each $F \in \Gamma$. In that case, with some extra effort, only $O(\log n)$ variables suffice to encode such flag systems. Finally, for any integer $x \geq 3$, it suffices to use only $O(|\Gamma|^{1/x})$ new variables if we are willing to replace the MN-encoding of an integer by words of length x.

8 Conclusion

The two major ideas in our construction is the "quadratic counter" as embodied in the Thue system S_0, and the technique of encoding flag systems. The obvious open question is whether the number of variables can be further reduced below $\sim 2n$.

The system S_2 can be trivially modified to carry out the Mayr-Meyer simulation of counter machines when the counters are known to be bounded by $e(n)$. We do not elaborate on this. (Of course, we could use S_3 or S_∞, but we prefer S_2 since it is explicitly given in this paper.) The other application of S_2, to show double exponential degree of reduced Gröbner bases along the lines of Huynh [3] or Mora-Möller [5] seems not so easy.[3] In the following, we give a simple proof that our system demonstrates the double-exponential degree behaviour for the ideal membership problem and syzygy problem. This kind of behaviour was first observed by Bayer-Stillman [1].

We use the following simple lemma from Mayr-Meyer [4]:

Lemma 4 *Given a commutative Thue system S over the variables Σ, let $F_S = \{f_1, \ldots, f_m\} \subseteq \mathbf{Q}[\Sigma]$ be the corresponding set of polynomials. Let $\beta, \alpha \in \Sigma^{\oplus}$. If*

$$\beta - \alpha = \sum_{i=1}^{m} g_i f_i \qquad (g_i \in \mathbf{Q}[\Sigma])$$

then there is a derivation

$$\alpha \longrightarrow w_1 \longrightarrow w_2 \longrightarrow \cdots \longrightarrow \beta (\mathrm{mod} S)$$

---

[3] Huynh provided only a sketchy justification of his rather involved construction; it seems non-trivial to nail down the necessary properties of such a system. Mora-Möller testify that their "proofs are too long to be included" in the paper. Both situations should be remedied in the sense of providing more transparent constructions and proofs.

such that for each j,

$$\deg(w_j) \leq \max\{\deg(g_i f_i) \ : \ i = 1, \ldots, m\}.$$

For the system (S_2, Σ_2), we now show:

Theorem 3 *Let* $F_2 = \{f_1, \ldots, f_m\} \subseteq \mathbf{Q}[\Sigma_2]$ *be the set of polynomials corresponding to the Thue system* S_2. *Let* $f_0 = Q_\infty - Q_0$. *Then* $f_0 \in (F_2)$. *Furthermore, for any sequence of polynomials* $\overline{g} = (g_1, \ldots, g_m)$, $g_i \in \mathbf{Q}[\Sigma_2]$, *if*

$$f_0 = \sum_{i=1}^{m} g_i f_i$$

then $\deg(\overline{g}) = e(n) - O(d)$.

Proof Clearly the derivation from Q_0 to $Q_\infty(\mathrm{mod} S_2)$ implies $f_0 \in (F_2)$. If $f_0 = \sum_{i=1}^{m} g_i f_i$, then the cited lemma of Mayr-Meyer implies a derivation $Q_0 \to w_1 \to w_2 \to \cdots \to Q_\infty$ where $\deg(w_j) \leq \max\{\deg(g_i f_i) \ : \ i = 1, \ldots m\}$ for each j. But this derivation, after omitting repetitions, is the standard derivation from Q_0 to Q_∞ and thus $\phi(w_\infty)$ occurs in this derivation. From $\deg(\phi(w_\infty)) \geq e(n)$ and $\deg(f_i) = d + O(1)$ $(i = 1, \ldots, m)$, the result follows. Q.E.D.

This double exponential lower bound for ideal membership in F_2 is easily turned into a similar bound for minimal syzygies. Consider the set of polynomials

$$F_2 \bigcup \{f_0\}.$$

Let $(g_0, g_1, \ldots, g_m) = \overline{g}$ be a minimal syzygy of $F_2 \bigcup \{f_0\}$ where $g_0 = 1$. Note that $\overline{g}$ exists. Hence $f_0 = -\sum_{i=1}^{m} g_i f_i$. Then the preceding result implies $\deg(\overline{g}) = e(n) - O(d)$.

9 Acknowledgements

I would like to thank Professors Helmut Alt, Bruno Buchberger and Emmerich Welzl for their warm hospitality during my sabbatical leave which made this work possible.

References

[1] David Bayer and Michael Stillman, "On the complexity of computing syzygies", to appear, *J. of Symbolic Computation.*

[2] Thomas W. Dubé, "The structure of polynomial ideals and Gröbner bases", forthcoming Courant Institute report, and N.Y.U. PhD thesis.

[3] Dung T. Huynh, "A superexponential lower bound for Gröbner bases and Church-Rosser commutative Thue systems", *Information and Control* 68 (1986) 196-206.

[4] Ernst W. Mayr and Albert R. Meyer, "The complexity of the word problems for commutative semigroups and polynomial ideals", *Advances in Mathematics* 46 (1982) 305-329.

[5] H. Michael Möller and Ferdinando Mora, "Upper and lower bounds for the degree of Gröbner bases", Eurosam 84, *Lecture notes in Computer Science No. 174*, 172-183.

[6] Emil L. Post, "Recursive unsolvability of a problem of Thue", *J. of Symbolic Logic* 12 (1947) 1-11.

NYU COMPSCI TR-416
Yap, Chee-Keng
A new lower bound construction for the word problem for... c.2

NYU COMPSCI TR-416
Yap, Chee-Keng
A new lower bound construction for the word problem for... c.2

DATE DUE	BORROWER'S NAME

This book may be kept

FOURTEEN DAYS

A fine will be charged for each day the book is kept overtime.

GAYLORD 142 PRINTED IN U.S.A.